

Future Trends for Semantic Wikis

Yaron Koren
Spring 2011 SMWCon
Arlington, VA

April 29, 2011

Some relevant long-term trends

- An explosion of data.
- Constant growth in the amount of RDF being produced.
- Growth in the amount of tools to process and visualize RDF.

SMW will (soon?) be able to save data
natively to an RDF triplestore.

I believe this represents the long-term
future of SMW's data storage.

Advantages of using an RDF triplestore (instead of a relational database)

- More scalable
- Allows for querying data from multiple sources
- Can be processed with generic tools

Currently, RDF triplestores are somewhat buggy (I've heard), and don't come pre-installed like MySQL and other DB systems do.

- This will presumably change in the future...

Current functions supported by SMW and related extensions

- Storage (SMW, Semantic Internal Objects)
- Editing (Semantic Forms, Halo, ...)
- Display (SMW, Semantic Result Formats, Semantic Maps, ...)
- Browsing (Semantic Drilldown, Semantic Treeview, ...)
- Import (Data Import, Data Transfer, External Data)
- Workflow (Semantic Tasks, Semantic Notifications, ...)

Which of these will still be necessary if
SMW moves to an RDF-based system?

Case in point: “Spark!” Javascript library

Co-created by our own Denny Vrandečić.

<http://km.aifb.kit.edu/sites/spark/>

A lightweight library to display results of SPARQL queries in standard display formats, like tables and charts.

There will be more like this in the future!

Once the technology is mature enough, there will be no need for a semantic wiki to be able to query its own data – it can use 3rd-party tools to do that.

What will still be needed?

- Storage (SMW, Semantic Internal Objects)
- Editing (Semantic Forms, Halo, ...)
- ~~Display (SMW, Semantic Result Formats, Semantic Maps, ...)~~
- ~~Browsing (Semantic Drilldown, Semantic Treeview, ...)~~
- *Import (Data Import, Data Transfer, External Data)*
(less important than before)
- ~~Workflow (Semantic Tasks, Semantic Notifications, ...)~~

Tools to edit pages' contents will remain
just as important.

This gets to the real strength of semantic
wikis: they're a framework for *text-based
structured data*.

Advantages of text: extremely flexible,
easy to do “diff”s

Not every semantic wiki application is text-based, but every semantic wiki application *should* be.

Future of text editing in structured / semantic wikis

Perhaps it, too, will use 3rd-party tools.

Possibly there will be a standard markup languages for forms – an outside library could take a form definition and a page's contents and display a form.

Other unknowns

- Will all wikis one day be semantic?
- Will all CMSES one day be wikis (i.e., keep a version history)?
- What will Wikipedia look like?